

**IFT1020 – TRAVAIL PRATIQUE #1 – 17 mai 2005****Le train entre en gare !**

Mohamed Lokbani

---

**Équipes:** Le travail est à faire en monôme (**une seule personne**).

**Remise :** Deux remises à effectuer : électronique et papier le **Judi 26 mai, 20h30 au plus tard, sans possibilité de retard**.

**Conseil:** N'attendez pas le dernier jour avant la remise pour commencer, vous n'aurez pas le temps.

---

**But :** Ce TP a pour but de vous faire pratiquer la procédure de remise, la rédaction d'un rapport, le respect des échéances et finalement mettre en pratique vos connaissances sur les applets.

**Enoncé :** On se propose d'écrire un programme Java permettant de dessiner un train. Ce programme porte le nom de « **AppTP1.java** » et il représente une applet.

**Description du train :** Le train est composé de 2 voitures et une locomotive à diesel. Il est en mouvement pendant une durée déterminée. Cette durée est simulée à travers une boucle for dont la valeur maximale est **T**. Pour chaque instant  $i$  ( $i < T$ ), on simule une pause qui correspond à un laps de temps d'attente **W**. Dès que ce temps d'attente est écoulé, le train va réapparaître (instant  $i+1$ ), à une position de départ (instant  $i$ ) plus un certain pas **P**. Cette façon de voir le train, équivaut à prendre des photos instantanées (une capture d'écran) du train à chaque  $x$  temps.

**Description de l'applet :** Pour amorcer cette applet, nous utilisons le fichier « **AppTP1.html** ». Ce dernier devra contenir les paramètres suivants :

**T** : la longueur de la boucle, par défaut : 80  
**W** : le délais d'attente, par défaut: 1000ms  
**P** : le pas de glissement, par défaut: 5 unités  
**BC** : la couleur de fond, par défaut verte  
**TC** : la couleur des voitures, par défaut rouge  
**FC** : la couleur de la fumée, par défaut noire

La fenêtre de dessin doit avoir une taille fixe de 400x200. Quand aux voitures, leur taille est de 80x30. Le diamètre des roues est de 80. La dimension de la locomotive est à calculer à partir des informations fournies. Et finalement, il est comme évident que le train ne peut circuler que sur des rails, qu'il faudra représenter aussi.

**Description des mouvements du train :** Les mouvements se font sur l'axe  $x$  uniquement. La position de départ est choisie de telle manière à faire apparaître le train en entier (voitures et locomotive). La position finale est fixée par la durée globale de la simulation (**T**). À noter que le calcul du délais d'attente entre chaque instant, doit utiliser une des fonctionnalités de la classe System et non pas une boucle for (ou while ou toute autre forme de boucle) à durée déterminée. Par ailleurs, il ne vous est pas permis d'utiliser les threads pour contrôler le temps mort (timeout, **W**).

Nous comptons sur votre imagination débordante pour ajuster cette applet à votre goût tout en tenant compte des contraintes imposées.

**Remise :** Il est important de noter que votre TP sera exécuté par les démonstrateurs sur une machine Linux avec le compilateur jdk (1.4.2). Si par choix vous décidez d'utiliser un autre compilateur, vérifiez que le code que vous avez produit (qui normalement fonctionne correctement chez vous) fonctionne bien sur les ordinateurs du DIRO. Avant de

faire cela, assurez vous d'abord que vous avez la bonne version de jdk, en utilisant pour cela la commande : "**java – version**" devra donnée le numéro de version "**1.4.2**".

La remise comprend **3 (TROIS)** choses (avant le jeudi 26 mai à 20h30) à Étienne :

1. Envoyez vos fichiers « AppTP1.java » et « AppTP1.html » par la procédure de remise électronique habituelle (Pour obtenir de l'aide sur cette commande, tapez dans un xterm : man remise). Respecter les noms des fichiers.

**remise dift1020 tp1** AppTP1.java AppTP1.html

2. Remettez une **copie papier** d'un rapport qui devrait décrire votre programme (pour le contenu d'un rapport voir la FAQ sur la page web du cours).
3. N'oubliez pas de remettre avec votre rapport une **copie papier** de votre programme Java.

### **Barème**

Il est à noter que votre programme doit être bâti sur une approche orientée objet. Ainsi donc, nous apporterons une grande attention à la manière avec laquelle vous avez conçu votre programme.

Ce TP1 est noté sur 4 points.

|                                                  |             |
|--------------------------------------------------|-------------|
| <b>Compilation et respect des spécifications</b> | <b>0.75</b> |
| <b>Codage, commentaires etc.</b>                 | <b>1.25</b> |
| <b>Rapport</b>                                   | <b>1</b>    |
| <b>Tests (fournis et non fournis)</b>            | <b>1</b>    |

En plus du précédent barème, vous risquez de perdre des points dans les cas suivants ....

- La non remise électronique (volontaire ou par erreur) est sanctionnée par la note 0.
- La non remise papier vous pénalise de 1 point.
- Les programmes ne contenant pas d'en-tête, -1 point.
- Un programme qui ne compile pas : 0/4.
- Un programme qui compile mais ne fait les choses prévues dans la spécification : 0/4.
- Le code doit être lisible, bien indenté et commenté.
- Le nom respect du nom de fichier, donc forcément il ne va pas compiler et un des points de la spécification n'a pas été respecté : 0/4.
- Aberration dans le codage : Même si tous les chemins mènent à Rome, faites l'effort nécessaire pour éviter de prendre le plus long! -0.25 et plus en fonction des aberrations détectées.

### **Des questions à propos de ce TP?**

L'adresse email: [dift1020@iro.umontreal.ca](mailto:dift1020@iro.umontreal.ca)

Pour faciliter le traitement de votre requête, inclure dans le sujet de votre email, au moins la chaîne: [IFT1020]

### **Mise à jour**

17-05-2005 diffusion